



Actividad 07 Practica 2

Alumno: José Osvaldo Farfán de León

Materia: Seminario de IA II

Profesor: Carlos Alberto Villaseñor Padilla

Sección: "D04"

Usando el Dataset de Covid, encuentra un buen clasificador no lineal para el problema.

Para iniciar con el problema Analice los datos del dataset y vi que los valores que teníamos guardados en las celdas eran de Yes y No por lo cual opte primero por cambiar estos valores a 1 y 0 para un análisis mas sencillo a mi forma de pensar:

```
1 import pandas as pd
2 from sklearn.preprocessing import LabelEncoder
3
4 data = pd.read_csv('Covid.csv')
5
6 label_encoder = LabelEncoder()
7
8 for column in data.columns:
9     data[column] = label_encoder.fit_transform(data[column])
10
11
12 data.to_csv('dataset_covid.csv', index=False)
```

Ahora procedemos a usar un algoritmo, en mi caso utilice el SVM para crear un clasificador lineal

Iniciamos cargando las bibliotecas que ocupare y el dataset descargado y previamente convertido a 1 y 0

```
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.svm import SVC
4 from sklearn.metrics import accuracy_score, confusion_matrix
5 from sklearn.preprocessing import StandardScaler
6 from sklearn.preprocessing import LabelEncoder
7
8 data = pd.read_csv('dataset_covid.csv')
9
```

Eliminamos la columna de salida y creamos nuestros valores de train y test

```
label_encoder = LabelEncoder()
data['COVID-19'] = label_encoder.fit_transform(data['COVID-19'])

X = data.drop('COVID-19', axis=1)
y = data['COVID-19']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Escalamos los datos:

```
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Creamos y entrenamos el SVM

```
classifier = SVC(kernel='rbf', C=1.0, gamma='scale')
classifier.fit(X_train, y_train)
```

Realizamos las predicciones así como también calculamos la precisión del modelo y mostramos la matriz de confusión

```
y_pred = classifier.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Precisión del modelo: {:.2f}%".format(accuracy * 100))

conf_matrix = confusion_matrix(y_test, y_pred)
print("Matriz de Confusión:\n", conf_matrix)
```

Obteniendo como resultado:

```
In [16]: runcell(0, '/home/farfan/Escritorio/Escuela/Actividad07 p2/code.py')
Precisión del modelo: 98.44%
Matriz de Confusión:
[[198  5]
 [ 12 872]]
```